IN THE CLAIMS

1. (Canceled)

2. (Canceled)

3. (Canceled)

4. (Canceled)

5. (Canceled)

6. (Canceled)

7. (Canceled)

8. (Currently Amended) The method as recited in claim 11, wherein the merging [[a]] the result comprises:

modifying contents of the second renamed register by placing data values of zero in the high-order bit positions of the second register;

adding contents of the first renamed register with the modified second renamed register; and

placing the result in the renamed result register.

9. (Canceled)

10. (Currently Amended) The method as recited in claim 11, wherein the merging [[a]] the result comprises:

modifying contents of the first renamed register by placing data values of zero in the low-order bit positions of the first renamed register;

modifying contents of the second renamed register by placing data values of zero in the high-order bit positions of the second renamed register;

adding the modified first renamed register with the modified second renamed register; and

placing the result in the renamed result register.

11. (Currently Amended) A method of performing sub-register data operations in executing an instruction[[]], the method comprising:

designating first and second operand registers and a result register, the result register being the same as one of the operand registers;

renaming each of the registers as a register within a plurality of registers such that each of the registers is a different register;

executing the instruction on a first renamed register and a second renamed register;

preventing the propagation of a carryover of a result of the executed instruction from low-order bit positions of a renamed result register to high-order bit positions of the renamed result register; and

merging the result of the executed instruction with a plurality of high-order bits from the first renamed register, the plurality of high-order bits being copied into the high-order bit positions of the renamed result register, and the result being placed into the low-order bit positions of the renamed result register.

12. (Previously Presented) The method of claim 11, wherein the first renamed register and the second renamed register have 32 bits.

13. (Previously Presented) The method of claim 11, wherein the renamed result register has 32 bits.

14. (Canceled)

15. (Previously Presented) The method of claim 11, wherein the result of the executed instruction is less than 32 bits.

16. (Original) The method of claim 15, wherein the result of the executed instruction is less than or equal to 16 bits.

17. (Original) The method of claim 16, wherein the result of the executed instruction is less than or equal to 8 bits.

18. (Previously Presented) The method of claim 11, wherein the merging a result is performed before instruction execution is complete.

19. (Currently Amended) A processor comprising:

an instruction set having an instruction;

first and second source registers and a destination register referenced by the instruction from the instruction set, one of said source registers and said destination register being the same;

a register renamer including:

a plurality of registers;

a lookup table linking first and second renamed source registers and a renamed destination register in said plurality of registers to said first and second source registers and said destination register, the renamed destination register being different than either of said first and second renamed source registers ; and

a logic circuit to examine the instruction before execution to identify a portion of one of said renamed source registers that should remain unchanged into the renamed destination register, and the logic circuit further to move the unchanged portion from one of the renamed source registers into the renamed destination register before instruction execution is complete, the logic circuit including a carryover circuit to prevent propagation of a carryover from the execution of the instruction to the unchanged portion of the renamed destination register.

20. (Currently Amended) The processor of claim 19, wherein the logic circuit is to move the unchanged portion into the renamed destination register by setting corresponding values of the one of the renamed source registers to zero.

21. (Currently Amended) The processor of claim 19, wherein the one of the renamed source registers and the renamed destination register have a greater bit-length than a result of the instruction.

22. (Currently Amended) The processor of claim 21, wherein the one of the renamed source registers and the renamed destination register each has 32 bits.

23. (Original) The processor of claim 21, wherein the result of the instruction has less than 32 bits.

24. (Original) The processor of claim 23, wherein the result of the instruction is less than or equal to 16 bits.

25. (Canceled)

26. (Canceled)

27. (Canceled)

28. (Currently Amended) A method comprising:

receiving an instruction to perform an operation on contents of first and second source registers and store the results of the operation in a destination register, the destination register being the same as one of the source registers, the contents including a plurality of bits and the operation results being a different bit length than bit lengths of the first and second source registers;

renaming the first and second source registers and the destination register as first and second renamed source registers and a renamed destination register, from among a plurality of registers, the renamed destination register being different than either the first or second renamed source register;

identifying high order bits of one of the renamed source registers that are to remain unchanged when merged into the renamed destination register;

modifying the contents of the other renamed source register by setting corresponding high order bits of the other renamed source register to zero;

adding the contents of the one of the renamed source registers with the modified contents of the other renamed source register;

placing results of the addition in the renamed destination register; and

preventing the propagation of a carryover of the addition results from low-order bit positions of the renamed destination register to high-order bit positions of the renamed destination register.

29. (Currently Amended) The method of claim 28, wherein screening the first and second renamed source registers comprises:

modifying contents of one of the renamed source registers by setting low order bits of the one of the renamed source registers to zero; and

modifying the contents of the other renamed source register by setting high order bits of the other renamed source register to zero.

30. (Previously Presented) The method of claim 29, wherein merging the operation results comprises:

adding the modified contents of the one of the renamed source registers with the modified contents of the other renamed source register; and

placing results of the addition into the renamed destination register.

31. (Currently Amended) A machine-readable medium having stored thereon a plurality of executable instructions for performing a method comprising:

receiving an instruction to perform an operation on contents of first and second source registers and store the results of the operation in a destination register, the destination register being the same as one of the source registers, the contents including a plurality of bits and the operation results being a different bit length than bit lengths of the first and second source registers;

renaming the first and second source registers and the destination register as first and second renamed source registers and a renamed destination register, from among a plurality of registers, the renamed destination register being different than either the first or second renamed source register;

identifying high order bits of one of the renamed source registers that are to remain unchanged when merged into the renamed destination register;

modifying the contents of the other renamed source register by setting corresponding high order bits of the other renamed source register to zero;

adding the contents of the one of the renamed source registers with the modified contents of the other renamed source register;

placing results of the addition in the renamed destination register; and

preventing the propagation of a carryover of the addition results from low-order bit positions of the renamed destination register to high-order bit positions of the renamed destination register.

32. (Original) The machine-readable medium of claim 31, wherein screening the first and second renamed source registers comprises:

modifying contents of one of the renamed source register by setting low order bits of the one of the renamed source registers to zero; and

modifying the contents of the other renamed source register by setting high order bits of the other renamed source register to zero.

33. (Original) The machine-readable medium of claim 32, wherein merging the operation results comprises:

adding the modified contents of the one of the renamed source registers with the modified contents of the other renamed source register; and

placing results of the addition into the renamed destination register.

34. (Currently Amended) A machine-readable medium having stored thereon a plurality of executable instructions for performing a method comprising:

designating first and second operand registers and a result register, the result register being the same as one of the operand registers;

renaming each of the registers as a register within a plurality of registers such that each of the registers is a different register;

executing the instruction on a first renamed register and a second renamed register;

preventing the propagation of a carryover of a result of the executed instruction from low-order bit positions of a renamed result register to high-order bit positions of the renamed result register; and

merging the result of the executed instruction with a plurality of high-order bits from the first renamed register, the plurality of high-order bits being copied into the high-order bit positions of the renamed result register, and the result being placed into the low-order bit positions of the renamed result register.

35. (Previously Presented) The machine-readable medium of claim 34, wherein the merging the result comprises:

modifying contents of the second renamed register by placing data values of zero in the high-order bit positions of the second renamed register;

adding the contents of the first renamed register with the modified second renamed register; and

placing the result in the renamed result register.

36. (Cancelled)

37. (Currently Amended) The machine readable medium of claim 36, wherein the merging the results comprises:

modifying the contents of the first renamed register by placing data values of zero in the low-order bit positions of the first renamed register;

modifying contents of the second renamed register by placing data values of zero in the high-order bit positions of the second renamed register;

adding the modified first renamed register with the modified second renamed register; and

placing the result in the renamed result register.

38. (Previously Presented) The method as recited in claim 11 wherein propagating a carryover result is accomplished by Anding the carryover result with a carry enable and said preventing propagation of a carryover result comprises removing said carry enable to prevent propagation of the carryover result.

39. (Currently Amended) The processor of claim 19 wherein said carryover circuit comprises a gate performing an Anding function having as inputs a carryover of [[the]] addition results from low-order bit positions of the renamed destination register and a carry enable and providing an input to high-order bit positions of the renamed destination register.

40. (Previously Presented) The method as recited in claim 28 wherein propagating a carryover result is accomplished by Anding the carryover result with a carry enable and said preventing propagation of a carryover result comprises removing said carry enable to prevent propagation of the carryover result.

41. (Previously Presented) The machine-readable medium of claim 31 wherein propagating a carryover result is accomplished by Anding the carryover result with a carry enable and said preventing propagation of a carryover result comprises removing said carry enable to prevent propagation of the carryover result.

42. (Previously Presented) The machine-readable medium of claim 34 wherein propagating a carryover result is accomplished by Anding the carryover result with a carry enable and said preventing propagation of a carryover result comprises removing said carry enable to prevent propagation of the carryover result.